AMENDMENTS TO THE CLAIMS

Claims 1 - 2 (Cancelled).

3. (New) In a microprocessor based on superscalar architecture capable of out-of-order execution, comprising: physical registers, the number of which is greater than that of the logical registers prescribed by the architecture; a free list that is designed to hold unallocated physical-register numbers; and a mapping table having entries that are provided in respective correspondence with the logical registers and that are each designed to hold a physical-register number; a method for performing register renaming in a pipelined manner, for each group of instructions that are to go through the process of register renaming simultaneously, comprising the steps of:

(a) tagging each logical-register number shown as a destination operand based on the order of the instructions in the group, and tagging each logical-register number shown as a source operand that is RAW (read-after-write) dependent on an instruction that goes through the process of register renaming simultaneously with the same tag as for the destination operand of said instruction; and

(b) renaming each tagged logical-register number to the physical-register number that is to be taken out of said free list and allocated in correspondence with the tag, and renaming each non-tagged logical-register number to the physical-register number that is to be obtained by accessing said mapping table.

4. (New) In a microprocessor based on superscalar architecture capable of out-of-order execution, comprising: physical registers, the number of which is greater than that of the logical registers prescribed by the architecture;

a free list that is designed to hold unallocated physical-register numbers; and a mapping table having entries that are provided in respective correspondence with the logical registers and that are each designed to hold a physical-register number; a method for performing register renaming in a pipelined manner, for each group of instructions that are to go through the process of register renaming simultaneously, comprising the steps of:

(a) in the cycle/cycles before the last cycle, tagging each logical-register number shown as a destination operand based on the order of the instructions in the group, and tagging each logical-register number shown as a source operand that is RAW (read-after-write) dependent on an instruction that goes through the process of register renaming simultaneously with the same tag as for the destination operand of said instruction; and

(b) in the last cycle, renaming each tagged logical-register number to the physical-register number that is to be taken out of said free list and allocated in correspondence with the tag, and renaming each non-tagged logical-register number to the physical-register number that is to be obtained by accessing said mapping table.